

conflict types;

allocating each conflict type a decision set which is used to indicate possible decisions which can be used to eliminate an inconsistency created by at least one operation of said respective conflict type; and

5 eliminating said inconsistency utilizing said decision set.

2. (Amended) The method as claimed in claim 1, further comprising the step of eliminating additional inconsistencies.

10 3. (Amended) The method as claimed in claim 1, further comprising the step of allocating each conflict type a decision set which is used to indicate possible decisions which can be used to eliminate an inconsistency created by additional operations of the respective conflict type.

15 4. (Amended) The method as claimed in claim 1, wherein said database collection contains a plurality of copy databases of said database.

5. (Amended) The method as claimed in claim 2, further comprising the step of ascertaining all inconsistencies and their dependencies on one another before said
20 step of eliminating said inconsistency.

6. (Amended) The method as claimed in claim 5, further comprising the step of ascertaining a conflict, an anomaly, or a pseudo-anomaly when an inconsistency is ascertained.

25 7. (Amended) The method as claimed in claim 2, further comprising the step of modifying, during elimination of said inconsistencies, said decision set for at least one conflict type depending on dependencies of said inconsistencies.

30

8. (Amended) The method as claimed in claim 2, further comprising the step of examining, after a prescribable number of eliminated inconsistencies, said database collection for further inconsistencies and their dependencies, anomalies and pseudo-anomalies.

9. (Amended) The method as claimed in claim 1, wherein said database collection contains an object-oriented database.

10. (Amended) The method as claimed in claim 1, further comprising the step of applying said method in a context of object-oriented software development.

11. (Amended) The method as claimed in claim 1, further comprising the step of applying said method in a context of creating a structured electronic document.

12. (Amended) An arrangement for eliminating at least one inconsistency in a database collection containing a database and at least one copy database of said database, which inconsistency arises on account of the database or said at least one copy database being changed, comprising:

a processor configured to:

allocate at least some operations which create an inconsistency to defined conflict types;

allocate to each conflict type a decision set which is used to indicate possible decisions which can be used to eliminate an inconsistency created by at least one operation of said respective conflict type; and

eliminate said inconsistency using said decision set.

13. (Amended) The arrangement as claimed in claim 12, wherein said processor is configured to eliminate a plurality of inconsistencies.

14. (Amended) The arrangement as claimed in claim 12, wherein said processor is configured to allocate each conflict type a decision set which is used to indicate possible decisions which can be used to eliminate an inconsistency created by a plurality of operations of said respective conflict type.

15. (Amended) The arrangement as claimed in claim 12, wherein said processor is configured to operate a database collection that contains a plurality of copy databases of said database.

16. (Amended) The arrangement as claimed in claim 13, wherein said processor is configured to ascertain all inconsistencies and their dependencies on one another before said inconsistencies are eliminated.

17. (Amended) The arrangement as claimed in claim 12, wherein said processor is configured to ascertain a conflict, an anomaly or a pseudo-anomaly when an inconsistency is ascertained.

18. (Amended) The arrangement as claimed in claim 13, wherein said processor is configured to modify, during elimination of said inconsistencies, a decision set for at least one conflict type depending on dependencies of said inconsistencies.

19. (Amended) The arrangement as claimed in claim 13, wherein said processor is configured to examine, after a prescribable number of eliminated inconsistencies, said database collection for further inconsistencies and their dependencies, anomalies and pseudo-anomalies.

20. (Amended) The arrangement as claimed in claim 12, wherein said processor is configured to operate on said database collection that contains an object-oriented database.